A Hybrid CBR-IR Approach to Legal Information Retrieval*

Edwina L. Rissland and Jody J. Daniels
Department of Computer Science
University of Massachusetts
Amherst, MA 01003

Abstract

In this paper we discuss a hybrid approach combining Case-Based Reasoning (CBR) and Information Retrieval (IR) for the retrieval of legal documents. Our hybrid CBR-IR approach takes as input a standard symbolic representation of a problem case and retrieves texts of relevant cases from a document corpus dramatically larger than the case base available to the CBR system. Our system works by first performing a standard HYPO-style CBR analysis and then using texts associated with certain important classes of cases found in this analysis to "seed" a modified version of INQUERY's relevance feedback mechanism in order to generate a query. Our approach provides two benefits: it extends the reach of CBR (for retrieval purposes) to much larger corpora, and it enables the injection of knowledge-based techniques into traditional IR. We describe our CBR-IR approach and report on on-going experiments performed in two different legal domains.

1 Introduction

There are many extensive and widely-used commercial text collections for use by those in the legal profession. For instance, all the cases decided in the Supreme Court and other Federal courts since their beginnings (in 1789) and in most state courts over at least the last 30 years are available through either West Publishing Company's WestLaw or Mead's Lexis systems. These massive on-line corpora represent a tremendous resource and investment of capital. They are the stock-in-trade of lawyers and others in the law, who use them extensively in legal research. While extensive, such commercial text-based systems are sometimes awkward to use and offer no guarantee for intelligent retrieval. The user of such a system must know how to manipulate them to get back truly relevant information.

*This research was supported by NSF Grant no. EEC-9209623, State/Industry/University Cooperative Research on Intelligent Information Retrieval, Digital Equipment Corporation, and the National Center for Automated Information Research.

For instance, it is often difficult to know exactly what terms to include in a query. Many cases about a topic, such as *offer*, do not explicitly contain the word *offer*, and just because a case does contain it, the case is not necessarily about this issue [Dick, 1987]. Often users of such systems are not even aware of the difficulties in using such systems because nothing has appeared to go wrong. One study found that although many users felt that they had retrieved a high proportion of the right documents (i.e., that recall was high), in fact they had retrieved only a mere 25% of the relevant texts [Blair & Maron, 1985].

The opposite problem of retrieving too much information, only some of which is actually relevant, is also a commonly occurring one. For example, if one were gathering precedents to be used in writing a brief for a personal (Chapter 13) bankruptcy case involving the legal question of court approval of the plan proposed by the debtor, WestLaw could be used to query its collection of bankruptcy cases, for instance, with the query *1325(a)* (the cite to the relevant section of the bankruptcy statute). Even with an additional restriction to cases decided between 1982 and 1990, this query produces 959 cases; far too many to be looked over by even the most dedicated legal researcher or research team. A more restricted query *1325(a)(3)* (the cite to the statutory subsection addressing the narrower "good faith" requirement for plan approval) retrieves 386 cases; still too many. Adding information about the case at hand (e.g., profession of debtor, amount of debts, duration of plan) or placing further restrictions on date and jurisdiction, would be ways to narrow down further the set of "good faith" cases retrieved.

By bringing in specifics of the case at hand—exactly the sort of information used by case-based reasoning (CBR) systems—it is possible to retrieve a workable set of truly relevant cases whose fact situations are similar, and not just those that happen to share a particular statutory cite. This is what an experienced user does and what vendors of such commercial systems recommend. In addition to facts of the current case, experienced users also draw on knowledge of known relevant precedents, past successful retrieval episodes on similar problems, knowledge of the particular domain, general information about courts and procedure, knowledge of how the retrieval engine works, etc. By being smart about query formation, a user can drive the retrieval engine to produce better results. However, it does require effort and expertise on the part of the user.

Thus we have two well-developed technologies, each with its own strengths and limitations. CBR can reason in depth about a problem case and, in particular, retrieve highly relevant cases, but this ability is limited by the availability of cases actually represented in a CBR system's case base. On the other hand, full-text information retrieval systems are not hampered by any lack of available cases (in textual form) but they cannot reason about a problem case and their sense of relevance is very weak.

A natural approach is to form a hybrid system where the strengths of each are used to overcome the weaknesses of the other in order to produce results or functionalities unachievable by either individually. Note that it is simply not realistic to think about re-vamping such text to suit the requirements of symbolic AI approaches, such as CBR. Such collections, built up over the years, will most likely be in their current textual form and be used pretty much as they are or not at all.

Our goal in this project is to take advantage of both the highly articulated sense of relevance used in CBR and the broadly applicable retrieval techniques used in IR in order to retrieve documents that are relevant to a problem case from commonly available large text bases without the need for creating a symbolic case representation for every document and without a lot of "driving" by the user. We require that the combination of CBR and IR be seamless: once the user has input a problem, the retrieval process should proceed *automatically*. In particular, the user should not be required to formulate queries.

Our hybrid CBR-IR approach takes as input a standard frame-based representation of a problem case (e.g., a case template filled by facts) and outputs texts of relevant cases retrieved from a document corpus many times larger than the case-base available to the CBR system. In one of our application domains, an area of tax law, the full-text collection is 500 times larger; in the other, an area of personal bankruptcy law, it is about 20 times larger. Since items in the larger document corpus are "represented" only in text form, they are not amenable to knowledge-based methods, in particular, indexing techniques used by CBR; thus they would ordinarily be unreachable by standard CBR. On the other hand, any form of knowledge-intensive reasoning of the kind at the core of CBR is not possible in text bases; a CBR-type definition of relevance is beyond the scope of traditional IR.

Our hybrid CBR-IR system works by first performing a standard HYPO-style CBR analysis [Ashley, 1990; Rissland & Ashley, 1987] and then using the results to cause the INQUERY IR system [Callan et al., 1992] to generate and act on a query in its usual way. In particular, our system causes a modified version of INQUERY's relevance feedback mechanism to generate terms and pairs of terms from the documents—for instance, full text opinions—associated with certain key cases found in the CBR analysis, such as most on-point cases. This use of relevance feedback, in effect, tells the IR component that the small set of "seed" cases found through the CBR analysis are highly relevant and that INQUERY should retrieve more like them.

Instead of the user initiating the retrieval by making up a query, in our approach the user begins by inputting facts of a case. Of course what the user gets back is a set of documents, not a nicely polished CBR analysis or argument; this is up to the user. However, the user has been able to perform an intelligent, problem-based retrieval from a large collection ordinarily beyond the reach of the CBR system.

In combining knowledge-based CBR with text-based IR, our approach allows the results of highly intelligent but small-scaled CBR to be highly leveraged to dramatically larger text collections, without the need for creating symbolic case representations for all documents in the collection. Our approach works to the benefit of both CBR and IR: it extends the reach of CBR and adds much-needed intelligence to traditional IR.

2 Background

Among current CBR systems there are few with large case-bases (say, larger than 1000 cases) and fewer still with both large case-bases and cases with in-depth representations [Kolodner, 1993]. All CBR systems use symbolic representations of cases and many—particularly, those developed in the legal domain—perform highly sophisticated reasoning. Even though CBR relieves the knowledge acquisition bottleneck by taking advantage of problem cases as they arise, it is still time-consuming to build a case corpus of significant size if cases are represented in any depth. If the case base is constructed after the fact from pre-existing archives of textual materials, the task can be daunting.

Most CBR systems that have represented large numbers of cases have used fairly simple case representations (e.g., MBRtalk [Stanfill & Waltz, 1986], PACE [Creecy et al., 1992], Anapron [Golding & Rosenbloom, 1991]) or have used representations easily derived from the problems they solve [Veloso, 1992]. In a very few situations, large case-bases have been constructed through a combination of case acquisition as a side-effect of customer service and follow-up knowledge engineering [Shimazu et al., 1993]. Our own CBR systems, which use detailed fact-oriented case representations—HYPO [Ashley, 1990; Rissland & Ashley, 1987], CABARET [Rissland & Skalak, 1991], BankXX [Rissland et al., 1993, 1994]—have typically had case bases in the range of three to five dozen cases.

Within the information retrieval (IR) world, there are huge document corpora and individual documents can be very large (e.g., tens of pages of text). However, the level of representation is shallow at best (i.e., the text itself) and the indexing is weak (e.g., based on statistics of the collection) [Salton, 1989]. Traditional IR systems suffer from a lack of knowledge about the domains, problems, uses, etc. of the information being dealt with. While IR systems use easily and broadly applied techniques, for instance, based on the occurrence of keywords or the frequency distribution of terms, these provide only the weakest sense of relevance, and users of IR systems make the implicit assumptions that not all the relevant documents will be retrieved (i.e., recall will not be perfect), and that not all of those retrieved are relevant (i.e., precision will not be perfect). CBR system users have higher expectations. We would like to extend our case-based retrieval to the IR context without sacrificing the high recall and precision associated with CBR and without enlisting the aid of an army of knowledge engineers to re-tool available text collections.

There have been several approaches for enriching retrieval environments with knowledge-based methods. Recently, Hafner and Wise used expert systems technology to help users pose requests to standard IR environments [Hafner & Wise, 1993]. Earlier work in legal conceptual information retrieval (e.g., [Hafner, 1987a, 1987b][Bing, 1987][Dick, 1987]) relied on a graph of diverse legal entities and concepts where labeled links captured influences and taxonomic information. A more recent project in legal information retrieval is Gelbart and Smith's FLEXICON [Gelbart & Smith,1991, 1993], which uses a vector space model for retrieval. FLEXICON can perform automatic thesaurus construction, relevance feedback, and can extract important paragraphs of an opinion to generate headnotes automatically.

Rose's SCALIR [Rose, 1994; Rose & Belew, 1991] is a hybrid symbolic/sub-symbolic system that uses a network of legal knowledge, including Shepard's links and West's key number taxonomy links, to perform retrieval. SCALIR uses spreading activation to perform the retrieval. Approximately 90% of the links in the SCALIR network are weighted connectionist links, with 75% of all the links between cases and terms.

In the FRANK project [Rissland et al., 1993], we explored how knowledge of the user's intended purpose for retrieving information—writing a one-sided pro-position advocacy brief, a balanced pro-con policy assessment memo, etc.—can be used to help configure CBR in order to retrieve useful cases. The high level purposes—the user's information needs—are used to specify what sort of cases to seek and which notions of similarity to use with the CBR.

In the BankXX project [Rissland et al., 1993, 1994], we explored the use of heuristic search as a program architecture for legal information retrieval. We represented components of argument at various levels of abstraction, for instance in so-called *argument pieces* and *argument factors*, and in various core components of the system, such as its evaluation function. These cause BankXX to search for, peruse, and possibly harvest information of known utility for making precedent-based arguments, such as ordinary and best pro and con cases, legal theories, factual prototypes. Such considerations can also be used to insure that the information BankXX retrieves is balanced in the sense that not all of it is cases, not all cases are for one side, etc.

In the CABARET project [Rissland & Skalak, 1991], we created a theory of statutory interpretation to guide not only the argumentative tasks pursued but also the type of cases retrieved in support of them. Our three-tier model of statutory argument—consisting of argument *strategies*, *moves*, and *primitives*—specified what types of cases are need to carry out a particular aspect of argument, such as broadening a rule by finding cases that do not satisfy certain statutory prerequisites but still were held to reap the benefit of the rule's conclusion (e.g., an allowed tax deduction).

A few other projects have tried to bridge the gap between CBR and IR. For instance, Aleven and Ashley performed some exploratory studies, as an off-shoot of their CATO project, on how knowledge of legal factors could be used in the formulation of natural language queries in Westlaw's WIN interface [Aleven & Ashley, 1993]. Specifically, they found that students taught to argue with factors can use them to produce good queries by expressing the factors in natural language (and then using WestLaw's WIN for retrieval). Their study did not involve the automatic generation of queries, but it would not be hard to do so. A potential problem, however, might be the limited number of terms in the query.

Goodman explored the opposite tack: enhance CBR with IR. This was done in the Prism system, a system for classifying bank telexes for further distribution and routing [Goodman, 1991]. Prism integrates IR methods for automatic index generation into the CBR paradigm. It uses a lexical pattern matcher to generate retrieval indices. Prism uses the retrieval indices to select cases from a case-base of over 9600 sample telexes. It then adapts the best matching cases to find classifications for the new telex.

3 Overview

Our system takes as input a problem case entered in the form of a generic case frame filled in with specific facts. It outputs a set of documents considered relevant to the problem case. See Figure 1.

Our system first uses its HYPO-styled CBR module to analyze a problem case with respect to the cases represented in its own "in-house" *case-knowledge-base* (*CKB*). In particular, it generates a *claim lattice*. This is done by sorting the CKB based on the intersection of each case's *dimensions* with those applicable in the problem case in the usual HYPO manner [Ashley, 1990; Rissland & Ashley, 1987]. Maximal cases in the resulting on-point ordering are called *most on-point cases* or *mopc's*; they are children of the root node, which contains the problem case, in the claim lattice.

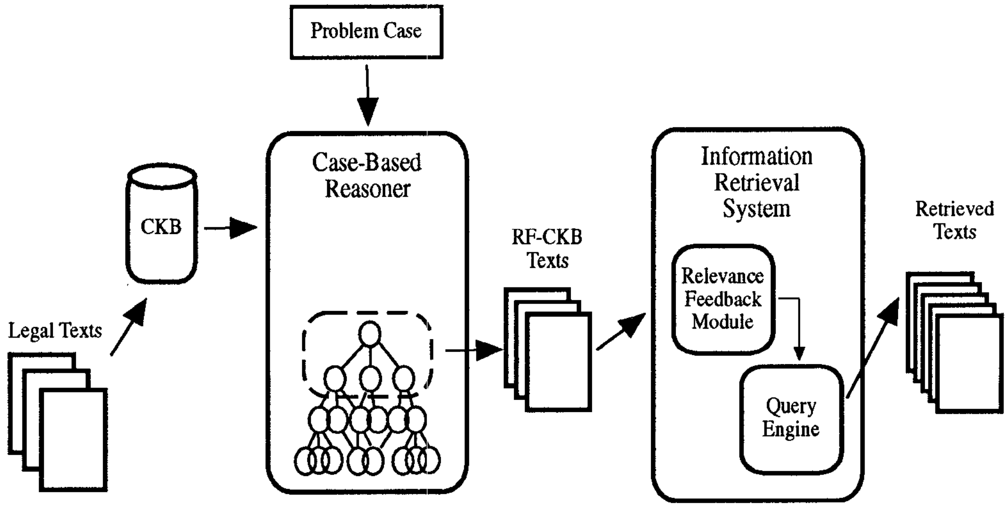

Figure 1. Overview of the hybrid CBR-IR architecture.

Next, from this claim lattice, the system selects a small number of certain special classes of cases, such as so-called *most on-point* cases (*mopc's*) or cases residing in the top two layers of the lattice.

The full-text versions of the opinions associated with these cases are then passed to the INQUERY retrieval engine, where a modified version of its relevance feedback (RF) mechanism is used to generate a query, consisting of the top *n* terms or pairs of terms. (Note, ordinarily INQUERY would not engage in relevance feedback until a retrieval, based on user input, had been made and the set of retrieved documents had been presented to the user for tagging as to their relevance.) The generated query is then submitted as usual.

INQUERY uses an inference network model, specifically a Bayesian probabilistic inference net [Turtle & Croft, 1991]. It uses a directed acyclic graph with the query or information need as the root node, document nodes as leaf nodes, and a layer of query concept nodes and a layer of concept representation nodes in between. It is also possible to have nodes representing complex query operators in a layer between the query nodes and the query concept nodes. This model allows INQUERY to combine multiple sources of evidence (beliefs) to determine relevance.

We call the sets of cases selected from the CBR's module analysis on its own CKB, and whose texts are submitted to the RF mechanism, the *relevance-feedback case-knowledge base* or *RF-CKB*. In this project, we have experimented with a variety of RF-CKB's.

Relevance feedback (RF) is a method for improving retrieval by having a user assess whether retrieved documents are relevant to his or her information needs. Using information derived from the texts tagged by a user as relevant, a relevance feedback algorithm alters the weights of the terms used in the original query, and/or adds additional query terms. The modified query is then submitted back to the IR engine. Relevance feedback has been found to improve precision by forty to sixty percent [Salton, 1989]. In our system, we employ relevance feedback on the cases from the RF-CKB with a null query.

4 Example

The following scenario illustrates our approach. Suppose a client consults with his lawyer about his attempt to take a tax deduction for an office in his home and the Internal Revenue Service has questioned it. The client believes that his deduction should be allowed. He tells his lawyer various facts concerning his situation. The lawyer inputs the case facts into the CBR-IR system. For example, assume the client is Mr. Weissman of the home office deduction case *Weissman v. Comm.*, 751 F.2d 512 (2d Cir. 1984).

Suppose the lawyer has knowledge of a set of home office deduction cases from her personal tax practice and that these make up the CKB used by the system. Assume these are cases from CABARET's case base. Using the lawyer's CKB, the CBR module next analyzes Mr. Weissman's case. Figure 2 shows cases in the resulting claim lattice. *Drucker, Gomez, Honan,* and *Meiers* are the mopc's.[1]

The combined CBR-IR system now uses this analysis to search for additional relevant cases within a larger corpus of legal texts, say those available through the WestLaw Federal Taxation Case Law collection. To do this, the system formulates a query by employing relevance feedback on the small set of RF-CKB cases selected from the claim lattice.

[1] *Weissman* is presented as an extended example in [Rissland & Skalak, 1991]. The claim lattice here is simpler than in that paper due to fewer cases used in the CKB.

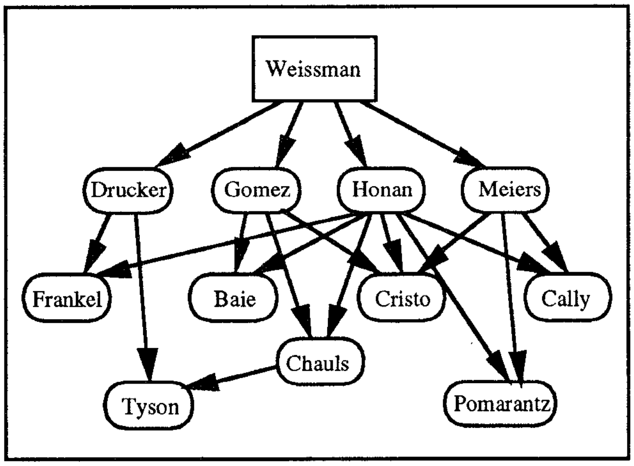

Figure 2. Claim lattice for the *Weissman* case.

Suppose the set of mopc's is used as the RF-CKB. The CBR system passes the indices for the texts (e.g., case opinions) associated with these mopc cases to the relevance feedback module within the INQUERY system, which then selects and weights the top terms, or pairs of terms from within these texts and forms a query. Finally, INQUERY acts on the query in the usual manner and returns a set of documents.

The lawyer can then use these cases (some of which she already knows about since they are in her own CKB) to do further research on Mr. Weissman's tax problem. Although not relieved of the need to peruse or study these additional cases, the lawyer has been able to access a large on-line document collection in a problem-specific manner without any need for formulating queries or otherwise cleverly manipulating the retrieval engine. Once the problem case is entered, the retrieval is done automatically by the system.

5 Methodology

In this section, we describe briefly the two domains of application, how we defined baselines and answer keys for our experiments, and the main parameters varied.

5.1 Domains

We have experimented with our approach in two domains thus far:

1 the *home-office domain.* used in CABARET

2 the *good faith bankruptcy domain*, used in BankXX

In this project, we did not design or create new CKB's. Rather, we used *as is* subsets of cases from CKB's developed in two past projects from our lab—CABARET [Rissland & Skalak, 1991] and BankXX [Rissland et al., 1994]—for the CKB's of our CBR-IR hybrid and for problem cases.

For the first domain, we used 25 cases from the original CABARET case base. CABARET's case base consisted of 36 real and hypothetical cases concerning the home office deduction, whose requirements are given in Section 280A(c)(1) of the Internal Revenue Code.

For the second domain, we used 45 of the 55 cases (those decided after 1981) from the original BankXX case base. BankXX's case base consisted of 55 actual cases concerning the "good faith" issue for the approval of debtor plans under Chapter 13 of the United States Bankruptcy code, (11 U.S.C. §§ 1301-1330) specifically in Section 1325(a)(3).

5.2 Problem Cases

In each domain, we ran a series of experiments by submitting a problem case, chosen from one of our CKB's. The system then treated it in a *de novo* manner: it is temporarily deleted from the CKB and treated as an entirely new problem situation with the remaining cases serving as the CKB. So far we have run experiments on 4 home office deduction cases and 3 bankruptcy cases as problem cases.

In the bankruptcy domain, problem cases were restricted to those cases from the BankXX corpus that were considered *meaty*, that is, they contained more than a set threshold of cited cases, theories, etc., in their opinions. We did this since so many of the cases in the BankXX CKB have sparse hand-coded answers, which can create evaluation problems [Rissland et al., 1995].

5.3 Building the Corpus

To test our approach, we constructed two test document collections to use in retrieval experiments:

1 The home office deduction domain corpus, called the *HOD-corpus*, consists of over 12,000 legal texts (opinions) from cases addressing a variety of issues.

2 The bankruptcy domain corpus, called the *Bankruptcy-corpus*, consists of over 950 legal texts (opinions) from cases addressing approval of a debtor's plan, as specified in Section 1325(a), which includes the "good faith" sub-issue in 1325(a)(3).

We built the HOD-corpus by adding approximately 200 cases, decided between 1986 and 1993, retrieved from the WestLaw Federal Taxation Case Law collection with the query *home office* to another already existing, nearly 12,000 document collection, called the *West collection*. The HOD-corpus includes 25 cases taken from CABARET's CKB. Of the 200 cases added in, only 128 actually concern the home office deduction. The original West collection probably contained no home office deduction cases.[2] Thus, only about 1% of the cases in the HOD-corpus address the home

[2]We had tested it with the query 280A, the relevant statutory section for the home office (and other) deductions, and only two texts were retrieved and neither addressed the home office deduction.

office deduction (280A(c)(1)) issue. It is a very diverse corpus.

We established a baseline for our experiments by using the simple one-term query *280A* to our HOD-corpus. This query does very well. It achieves 81.1% average precision (calculated on the basis of 11 precision-recall points). This is a high baseline to improve on.

On the other hand, the Bankruptcy-corpus contains cases dealing only with the aspects of debtor plan approval given in Section 1325(a). We built this corpus by downloading all 962 cases decided between 1982 and 1990 that were found with the query *1325(a)* to the WestLaw Federal Bankruptcy Case Law collection. It contained all but the 10 earliest cases from the original 55-case BankXX CKB. In the Bankruptcy-corpus about 40% (385 cases) make specific reference to the phrase *good faith*. This corpus is very focused.

We established a baseline for this corpus by using the simple one-phrase query *good faith* on our Bankruptcy-corpus. This results in an average precision of 89.3%. This high value indicates that a high proportion of "good faith" cases actually use that phrase and that cases on other 1325(a) issues do not. This is an exceedingly high baseline.

We found that home office deduction cases often discuss more than just the home office deduction 280A(c)(1) issue. As many as seven or more other issues might be covered within such a case. On the other hand, we found that most of our bankruptcy cases only addressed the one "good faith" (1325(a)(3)) issue. Not surprisingly, the home office deduction cases vary significantly in length—anywhere from one to twenty or more pages in length—whereas the bankruptcy cases tend to be on the shorter side, running generally less than ten pages. Thus Bankruptcy cases are much more concise and focused than home office deduction cases.

5.4 Answer Keys

For each problem, we constructed an "answer key" that specified the documents to be considered as relevant. In the home office deduction domain, any of the 128 cases from the HOD-corpus that actually concerns a taxpayer trying to take the home office deduction is considered relevant. In the bankruptcy domain, any case that addresses the "good faith" issue is considered relevant. Thus all problem cases were assigned the same set of texts as the correct answer, which includes those which CABARET or BankXX would have considered relevant. We plan to use a more refined sense of correct answer—those actually cited in the court case—in future work.

Answer keys are used to calculate precision, recall, and average precision statistics:

•***Recall*** measures the percent of those items that should have been retrieved by the query that actually were. It measures coverage. It is the ratio of the number of relevant retrieved items (i.e., items in the intersection of the answer key and the retrieved items) to the total number of relevant items

•***Precision*** measures the percent of retrieved items that are relevant. It measures accuracy. It is the ratio of the number of relevant retrieved items to the total number of retrieved items.

•***Average precision*** is the average of the precision values achieved at 11 levels of recall: 0%, 10%, 20%, ... 100%.

6 Experiments

In this section, we discuss our experiments with the hybrid CBR-IR approach. In particular, we discuss results achieved with different RF-CKB's and different numbers of terms used in the resulting query. In another report, we discuss experiments using *pairs of terms* [Daniels & Rissland, 1995].

6.1 System Parameters Varied

For each problem case, we varied the following aspects of the CBR-IR system:

1. the RF-CKB used to seed the relevance feedback mechanism;
2. the number of terms used in the INQUERY query.

We did not vary other parameters used in relevance feedback, such as the weighting metric. For our experiments, there is no "original query" *per se*. Instead, the relevance feedback module is given a null query and a small number of legal texts, the RF-CKB, as its set of relevant documents. Because there is no original query, some aspects of relevance feedback, such as re-weighting of terms, do not apply.

The RF module of INQUERY calculates the top terms within an RF-CKB, weights them, and submits them as a "new" query against the collection. It uses only terms found from within the RF-CKB. For each RF-CKB, the relevance feedback module formed a query with the top 5, 10, 15, 20, 25, 50, 100, 150, 200, 250, 300, 350, and 400 terms found in the RF-CKB. The maximum length query was 400 terms, because of a limitation of the RF module. Therefore longer queries, such as all of the terms from within a RF-CKB, were not tested.

6.2 RF-CKB's—Sets of Cases for Seeding Relevance Feedback

For the home office deduction domain, we selected 4 cases to use as problem cases. On *Weissman*, the first problem case with which we experimented, we examined the queries and resulting precision-recall results (see Figure 4) derived with six different RF-CKB's:

1. **RF-CKB1** consists of the mopc's. In *Weissman*, there are 4 mopc's. (See Figure 2.) Coincidentally, this set of four texts happens to be *pure* in the sense that there are no other issues under consideration in them besides that of the home office deduction. An *impure* case discusses the home office deduction and one or more other issues.[3]

2. **RF-CKB2** consists of only *impure* cases; a random selection of five of them from the claim lattice. RF-CKB2 tests the ability of relevance feedback in discriminating important terms from non-relevant ones within noisy texts.

3. **RF-CKB3** is the union of RF-CKB1 and RF-CKB2. It has both pure and impure texts and is thus *mixed*. RF-CKB3 has the advantage of having a large number terms from which to select the important ones.

4. **RF-CKB4** contains all eight pure texts from the top two layers of the claim lattice. It is comprised of the four mopc's and four more cases.

5. **RF-CKB5** contains all seven impure texts in the CBR module's CKB of 25 cases.

6. **RF-CKB6** contains all the cases in the top 2 layers of the claim lattice. It contains 11 cases: eight pure texts (RF-CKB4) and three impure.

	West	RF-CKB1	RF-CKB2	RF-CKB3	RF-CKB4	RF-CKB5	RF-CKB6
Documents	11953	4	5	9	8	7	11
Unique Terms	142749	1242	2430	2885	1952	2941	2767
Unique Terms per Text (avg)	530	477	842	680	516	834	59
Avg Length	3250	1254	3321	2402	1533	3353	2031

Figure 3. RF-CKB statistics for the Home Office Deduction experiments.

On average, the pure texts are much smaller than a typical text in the West collection. Not surprisingly, across all RF-CKB's, pure cases had the least number of unique terms to select among and tended to be shorter documents, while the impure documents had much larger numbers of unique terms and were significantly longer. The average West document was as long as the impure, yet had significantly fewer unique terms. See Figure 3.

[3]Of the 25 cases in the CBR module's CKB, 7 are not pure. Among the other 103 home office deduction cases in the HOD-corpus, fewer than 10 were pure.

7 Results

Eleven point precision and recall tables were generated for each query. Figure 4 gives a summary of the average precisions for the six RF-CKB's used on the *Weissman* case with different numbers of terms used to form a query. RF-CKB1 containing pure mopc's requires the most terms—between 51 and 100 terms—to achieve an average precision exceeding the baseline of 81.1%. The small impure RF-CKB2 achieves this average between 11 and 15 terms and the mixed RF-CKB3 needs 5 or less terms. Overall, the top two layers RF-CKB6 achieves the best set of average precisions and the 8 pure RF-CKB4 next; the 7 impure RF-CKB5 achieves the worst.

Every RF-CKB results in significant improvement over the baseline average precision of 81.1% by the time they have included 100 or fewer terms. The mixed RF-CKB3 does this with just 5 terms. Relative improvement over the baseline is nearly 10% in many queries. Thus, the hybrid CBR-IR method significantly outscores straight IR alone.

There is a large jump in the average precisions for most of the RF-CKB's. For example, for RF-CKB1 a jump from 36.3% to 79.3% occurs between 16 and 20 terms. For the small impure RF-CKB2, the jump is from 54.0% to 88.1% with the addition of terms 11 to 15. Jumps may be explained by examining the set of terms that are added to the longer queries. It turns out that whenever the jump occurs, both *280A* and *dwell* are new terms. No such large jump is apparent with the mixed RF-CKB3, since both terms are used in all the queries, from 5 terms on up.

We had expected that the mopc/pure RF-CKB1 would outperform the other RF-CKB's. Its failure to do so may be due to its small number (4) of small documents from which the RF mechanism generates terms. By contrast, for instance, the mixed RF-CKB3 had twice as many documents (9) and the average size of its documents is approximately twice that in RF-CKB1. In the impure RF-CKB2, the average document is more than two and a half times as large. The number of unique terms in the RF-CKB1 is also significantly smaller than the average found in RF-CKB2 and RF-CKB3.

In addition, the ability of the RF mechanism to select high-value terms may be restricted by the purity of the texts in the mopc/pure RF-CKB1. The four cases in RF-CKB1 discuss only issues surrounding taking the home office deduction. Therefore, it should follow that these documents contain terms descriptive for the home office deduction.

Number of Terms	RF-CKB1 Mopc/ Pure	RF-CKB2 5 Impure	RF-CKB3 Mixed	RF-CKB4 8 Pure	RF-CKB5 7 Impure	RF-CKB6 Top 2 Layers
5	40.6	55.2	**83.8**	39.5	53.1	39.9
10	38.6	54.0	**86.7**	42.5	63.8	**83.8**
15	36.3	**88.1**	**86.5**	**83.0**	66.8	**83.7**
20	79.3	**90.7**	**86.3**	**83.1**	68.4	**85.3**
25	79.0	**87.6**	**88.8**	**83.8**	68.1	**89.0**
50	78.9	**87.5**	**89.3**	**88.1**	**85.7**	**89.0**
100	**81.2**	**87.5**	**88.5**	**88.5**	**83.5**	**90.3**
150	**85.9**	**87.5**	**88.4**	**89.0**	**83.5**	**90.2**
200	**86.6**	**88.2**	**88.4**	**88.9**	**83.5**	**90.2**
250	**87.4**	**86.5**	**88.3**	**89.2**	**83.6**	**90.5**
300	**87.6**	**86.5**	**89.2**	**89.2**	**82.0**	**90.2**
350	**86.4**	**86.0**	**89.1**	**88.5**	80.7	**89.8**
400	**85.4**	**85.4**	**88.8**	**88.8**	**81.9**	**89.3**

Figure 4. For the top *n* terms, average precision achieved by RF-CKB's on *Weissman*. Boldface indicates the query exceeded the baseline of 81.1%

Yet, because so many terms occur across all four relevant documents, the high-value terms may be hard to discriminate and are thus undervalued by the RF mechanism. Discriminating high-value terms within impure and mixed RF-CKB's might be more easily done than within this pure RF-CKB. Within an impure RF-CKB, terms descriptive of the home office deduction comprise a smaller proportion of each text because additional issues are represented. This may aid the selection metric in finding terms descriptive of the home office deduction. Within the mixed RF-CKB3, the impure documents may provide the noise necessary for these high-value terms to be more recognizable.

Based on the results from these experiments, we ran a similar set of experiments for three other cases from the HOD domain, *Honan*, *Meiers*, and *Soliman*[4] using just two RF-CKB's:

1. RF-CKB1: the set of mopc's for a problem case.
2. RF-CKB6: the top two layers of a problem case's claim lattice.

These results were similar to those found with *Weissman*. All the mopc RF-CKB's exceeded the baseline by 100 terms or less. Using the top two layers exceeded the baseline within 10 or fewer terms and achieved better overall results than the mopc RF-CKB's.

Within the bankruptcy domain we selected three problem cases and used these two same RF-CKB's. The Bankruptcy term results were not as spectacular. The system was able to achieve average precisions ranging only from 48% to 67%. Again, better average precision occurs with higher numbers of terms and again, RF-CKB6 composed of the top two layers of the lattice outperformed RF-CKB1, composed of only mopc's. Random sets of four or five documents achieved average precisions in approximately the same range. It should be noted that the total number of documents used by the relevance feedback module was still very small; the largest RF-CKB contained only nine documents.

It also should be noted that in the experiments reported here, we restricted our queries to **simple terms** whereas the baseline used a query composed of a **phrase**. Phrases can be much more descriptive of a text's content. This explains why we did not outscore the exceedingly high baseline average precision (89.3%) of the bankruptcy corpus. In another set of experiments using phrases in the bankruptcy domain, we demonstrate that phrases outperform single terms across all of our RF-CKB's. In particular, we did outperform the baseline with phrases [Daniels & Rissland, 1995].

8 Conclusion

The goal of this project is to create a system that provides access to more cases than usually provided by a CBR system and with a more precise sense of relevance than provided by traditional IR systems. In our hybrid CBR-IR approach, knowledge-intensive case-based reasoning is performed on a (small) corpus of cases represented in a CBR module, and important cases selected from this analysis are used to drive a traditional text-based IR system using a (much) larger document collection. We use the CBR analysis to locate good examples of the kind of cases we want, and the IR system to retrieve more like them.

In this paper, we have explored how a limited number of relevant full-text legal documents can be used to retrieve—with a high level of both recall and precision—additional relevant legal texts from a large corpus. We have shown that using a modified version of relevance feedback, in which we have no initial query to modify and a small

[4] *Honan v. Comm.*, T.C. Memo. 1984-253; *Meiers v. Comm.*, 782 F.2d 75 (7th Cir. 1986); *Soliman v. Comm.*, 935 F.2d 52 (4th Cir. 1991).

number of well-chosen full-text documents, we can do this *automatically* with excellent results.

In summary, our approach integrates CBR with IR to:

- extend the range of retrievals to materials outside the scope of a CBR system
- improve the precision of traditional IR
- leverage the strengths of each
- achieve robust, decent results with minimal effort
- require no human in the loop, other than case entry
- be reproducible across a variety of problem cases.

In general, our system achieves best results when 150 or more terms are used. The system achieves good results even if it is restricted to a small set of short texts that all discuss the same issue. Furthermore, using a large number of terms (300-400) does not degrade performance as much as might be expected, and, in fact, in most instances achieved results as good as or better than queries with fewer terms.

We are now examining the use of pairs of terms found in close proximity to each other to be used as the query terms. Additionally, we are conducting experiments using a more refined problem-specific sense of relevance: a case is considered relevant only if it is actually *cited* in the actual opinion of the problem case. We are able to use this stricter definition of relevance because we already have hand-coded answers for individual problems available from our empirical evaluation of the BankXX system [Rissland et al., 1995], in which we compared the sets of items (e.g., cases, legal theories) retrieved by BankXX against those actually mentioned in a case.

References

Ashley, K. D. (1990). *Modeling Legal Argument: Reasoning with Cases and Hypotheticals.* M.I.T. Press, Cambridge, MA. 1990.

Aleven, V., & Ashley, K. D. (1993). What Law Students Need to Know to WIN. *Proceedings of the Fourth International Conference on Artificial Intelligence and Law (ICAIL-93)*, 152-161. Amsterdam. ACM Press.

Bing, J. (1987). Designing Text Retrieval Systems for "Conceptual Searching". *Proceedings of the First International Conference on Artificial Intelligence and Law (ICAIL-87)*, 43-51. Boston. ACM Press.

Blair, D. C., & Maron, M. E. (1985). An Evaluation of Retrieval Effectiveness for a Full-Text Document-Retrieval System. *Communications of the ACM*, 28(3): 289-299. March 1985.

Callan, J. P., Croft, W., B., & Harding, S. M. (1992). The INQUERY Retrieval System. In A.M. Tjoa and I. Ramos (Eds.), *Database and Expert Systems Applications: Proceedings of the International Conference*, 78-83. Valencia, Spain. Springer Verlag.

Creecy, R. H., Masand, B. M., Smith, S. J., & Waltz, D. L. (1992). Trading MIPs and Memory for Knowledge Engineering. *Communications of the ACM*, 34(8), 48-64, August 1992.

Daniels, J. J., & Rissland, E. L. (1995). A Case-Based Approach to Intelligent Information Retrieval. To appear in *Proceedings of the 18th Annual International Conference on Research and Development in Information Retrieval (SIGIR-95)*, Seattle, July 1995. ACM Press.

Dick, J. (1987). Conceptual Retrieval and Case Law. *Proceedings, First International Conference on Artificial Intelligence and Law (ICAIL-87)*, 106-115. Boston. ACM Press.

Gelbart, D. & Smith, J.C. (1991). Beyond Boolean Search: FLEXICON, A Legal Text-Based Intelligent System. *Third International Conference on Artificial Intelligence and Law (ICAIL-91)*, 225-234. Oxford. ACM Press.

Gelbart, D. & Smith, J.C. (1993). FLEXICON: An Evaluation of a Statistical Ranking Model Adapted to Intelligent Legal Text Management. *Proceedings of the Fourth International Conference on Artificial Intelligence and Law (ICAIL-93)*, 142-151. ACM Press.

Golding, A. R. & Rosenbloom, P. S. (1991). Improving Rule-based Systems Through Case-Based Reasoning. *Proceedings, Ninth International Conference on Artificial Intelligence (AAAI-91)*, 22-27. Anaheim, July 1991.

Goodman, M. (1991). Prism: A Case-based Telex Classifier, In Alain Rappaport and Reid Smith, (Eds.) *Innovative Applications of Artificial Intelligence—2*, 25-37. AAAI Press, Menlo Park, CA.

Hafner, C. D. (1987a). Conceptual Organization of Case Law Knowledge Bases. *Proceedings, First International Conference on Artificial Intelligence and Law (ICAIL-87)*, 35-42. Boston. ACM Press.

Hafner, C. D. (1987b). *An Information Retrieval System Based on a Computer Model of Legal Knowledge*. Ph.D. thesis, University of Michigan. Republished by UMI Research Press, Ann Arbor, MI (1981).

Hafner, C. D., & Wise, V. J. (1993). SmartLaw: Adapting "Classic" Expert System Techniques for the Legal Research Domain. *Proceedings of the Fourth International Conference on Artificial Intelligence and Law (ICAIL-93)*, 133-142. Amsterdam. ACM Press.

Kolodner, J. L. (1993). *Case-Based Reasoning*. Morgan Kaufmann.

Rissland, E. L., & Ashley, K. D. (1987). "A Case-Based System for Trade Secrets Law." *Proceedings First International Conference on AI and Law, (ICAIL-87)*, 60-66. Boston. ACM Press.

Rissland, E. L., Daniels, J. J., Rubinstein, Z., & Skalak, D. B. (1993). "Case-based Diagnostic Analysis in a Blackboard Architecture." *Proceedings of the Eleventh National Conference for Artificial Intelligence (AAAI-93)*, 66-72. Washington, D.C., July 1993.

Rissland, E. L., & Skalak, D. B. (1991). "CABARET: Statutory Interpretation in a Hybrid Architecture." *International Journal of Man-Machine Studies (IJMMS)*, June, 1991, (34): 839-887.

Rissland, E. L., Skalak, D. B. & Friedman, M. T. (1993). BankXX: A Program to Generate Argument through Case-Based Search. *Proceedings Fourth International Conference on Artificial Intelligence and Law (ICAIL-93)*, 117-124. Amsterdam. ACM Press.

Rissland, E. L., Skalak, D. B. & Friedman, M. T. (1994). *BankXX: Supporting Legal Arguments through Heuristic Retrieval.* TR 94-76, Dept. of Computer Science, Univeristy of Massachusetts, Amherst. Submitted for publication.

Rissland, E. L., Skalak, D. B. & Friedman, M. T. (1995). *Evaluating a Legal Argument Program: The BankXX Experiments.* TR 95-30, Dept. of Computer Science, Univeristy of Massachusetts, Amherst. Submitted for publication.

Rose, D. E. & Belew, R. K. (1991). A Connectionist and Symbolic Hybrid for Improving Legal Research. *International Journal of Man-Machine Studies,* 35, 1-33.

Rose, D. E. (1994). *A Symbolic and Connectionist Approach to Legal Information Retrieval.* Lawrence Erlbaum.

Salton, G. (1989). *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer.* Addison-Wesley.

Shimazu, H., Kitano, H., & Shibata, A. Retrieving Cases from Relational Data-Bases: Another Stride Toward Corporate-Wide Case-Base Systems. *Proceedings 13th International Joint Conference on Artificial Intelligence (IJCAI-93),* 909-914. Chambery, France. Morgan-Kaufmann.

Stanfill, C., & Waltz, D. L. (1986). Toward Memory-Based Reasoning. *Communications of the ACM*, 29(12):1213-1228. December 1986.

Turtle, H. R., & Croft, W. B. (1991). Evaluation of an Inference Network-Based Retrieval Model. *ACM Transactions on Information Systems*, 9(3): 187-222. March 1991.

Veloso, M. M. (1992). *Learning by Analogical Reasoning in General Problem Solving.* Ph.D. Thesis, Carnegie Mellon University, Pittsburgh, PA.